

The submodular secretary problem under a cardinality constraint and with limited resources

Tom Hess and Sivan Sabato
Ben-Gurion University of the Negev
Beer Sheva, Israel

Abstract

We study the submodular secretary problem subject to a cardinality constraint, in long-running scenarios, or under resource constraints. In these scenarios the resources consumed by the algorithm should not grow with the input size, and the online selection algorithm should be an anytime algorithm. We propose a 0.1933-competitive anytime algorithm, which performs only a single evaluation of the marginal contribution for each observed item, and requires a memory of order only k (up to logarithmic factors), where k is the cardinality constraint. The best competitive ratio for this problem known so far under these constraints is $\frac{e-1}{e^2+e} \approx 0.1700$ (Feldman et al., 2011). Our algorithm is based on the observation that information collected during times in which no good items were selected, can be used to improve the subsequent probability of selection success. The improvement is obtained by using an adaptive selection strategy, which is a solution to a stand-alone online selection problem. We develop general tools for analyzing this algorithmic framework, which we believe will be useful also for other online selection problems.

1 Introduction

We study the Submodular Secretary Problem subject to a cardinality constraint. In the classical Secretary Problem (Dynkin, 1963; Gilbert and Mosteller, 1966), n items appear at a random order, each with an attached value. The algorithm is allowed to select a single item. If it decides to select an item, it must do so immediately, before observing the next items, and it cannot later change its decision. The goal of the algorithm is to select the maximal-value item with the highest probability, where the set of items is selected by an adversary and the order of their appearance is random.

In the Submodular Secretary Problem, introduced by Gupta et al. (2010) and Bateni et al. (2013), the algorithm is allowed to select more than one item, and its goal is to maximize the value of the set of selected items. The value of the set is given by a monotone submodular function. We study the submodular secretary problem subject to a cardinality constraint, in which the number of elements that the algorithm is allowed to select is bounded. The goal is to be competitive with the optimal (offline) algorithm: An online algorithm has a competitive ratio of c if for any universe of items of any size n , any cardinality constraint k , and any monotone submodular function f , $\mathbb{E}[f(\text{ALG})] \geq c \cdot f(\text{OPT})$, where OPT is the subset of the input that maximizes f under the cardinality constraint, and ALG is the set of items selected by the algorithm.

In this work we consider the submodular secretary problem in long-running scenarios, or under resource constraints. In many scenarios (e.g., web-servers and interactive applications) online algorithms run for a long time or without a known end time. In these scenarios the resources consumed by the algorithm should not grow with the input size, and the online selection algorithm should be an *anytime* algorithm (Dean and Boddy, 1988): The set selected by the algorithm should be competitive with the best offline solution available so far at (almost) any time during the run, and not only at the end of the run. To this end, we propose an algorithm which performs, for each observed item, a single evaluation of the item's marginal contribution to the value of f , and requires a memory of order k only (up to logarithmic factors).¹

¹Order k memory is necessary only for storing previously selected items, for the sake of evaluating the marginal contribution to f of new items. If these marginal evaluations are provided by an external oracle, then our the memory required by our algorithm is $O(\log(n/k))$.

Recently, Kesselheim and Tönnis (2016) proposed an algorithm with a competitive ratio of 0.238 for the submodular secretary proposed under a cardinality constraint. However, this algorithm uses the offline greedy algorithm as a black box, and this black box is executed for every observed item. This approach requires (after reasonable optimization) $\Omega(nk^2)$ evaluations of the submodular function, and $\Omega(n)$ memory. These resource requirements could be prohibitive in many applications if either n or k are large. In addition, the proposed algorithm does not select any element until observing approximately n/e items, thus it does not have the anytime property.

Our contribution. Our propose algorithm, CSS, satisfies the required constraints, and has a competitive ratio of 0.1933. The best competitive ratio for this problem known so far that satisfies these constraints is $\frac{e-1}{e^2+e} \approx 0.1700$ (Feldman et al., 2011). CSS is based on the observation that information collected during times in which no good items were selected, can be used to improve the subsequent probability of selection success. The improvement is obtained by using an adaptive selection strategy, which is a solution to a stand-alone selection problem described in Section 4. We develop general tools for analyzing this algorithmic framework. A main challenge is to lower bound the competitive ratio when the selection success probability varies with time and depends on previous events. We show that the algorithm’s behavior can be modeled as a Markov chain, and use Markov theory to derive a competitive ratio lower bound. We believe that this approach will be useful also for other online selection problems.

Structure of paper. Preliminaries are given in Section 2. The main results are introduced in Section 3. The proposed algorithm, CSS, is given in Section 4. CSS is analyzed and the main result is proved in Section 5. For simplicity, the full algorithm and the analysis are provided for the case of n larger than some constant n_0 . In Section 6 we discuss how to adapt the algorithm to small n without violating the resource constraints. We conclude in Section 7. Some proofs are deferred to Appendix A.

Related work

Many variants have been introduced for the classical Secretary Problem. The matroid Secretary Problem was introduced in Babaioff et al. (2007b). In this problem the selected set should be an independent set of a matroid and the value function is modular. Several works show constant competitive ratios for specific matroids, (e.g., Babaioff et al., 2007b; Dimitrov and Plaxton, 2008; Korula and Pál, 2009; Feldman et al., 2011; Gupta et al., 2010; Bateni et al., 2013). For the uniform matroid, which is equivalent to a cardinality constraint, and a modular value function, Kleinberg (2005) proposed an algorithm with a competitive ratio of $1 - \frac{5}{\sqrt{k}}$ and Babaioff et al. (2007a) achieved a ratio of $1/e$ for all k . For a general matroid, the best known competitive ratio is $O(\log \log k)$, where k is the rank of the matroid (Feldman et al., 2015; Lachish, 2014).

Offline Submodular maximization under matroid constraints has been studied, for instance, in Calinescu et al. (2011); Krause and Golovin (2012). The Submodular Secretary Problem with a matroid constraint was introduced in Gupta et al. (2010) and independently in Bateni et al. (2013). Feldman and Zenklusen (2015) show a reduction from the submodular matroid secretary problem to the modular matroid secretary problem. The best competitive ratio for the submodular secretary problem under a cardinality constraint, which satisfies the resource constraints described above, achieves a competitive ratio of $\frac{e^{-1}(1-e^{-1})}{1+e^{-1}} \approx 0.1700$ (Feldman et al., 2011). A competitive ratio of 0.238, with an algorithm that does not satisfy these constraints, was recently shown in Kesselheim and Tönnis (2016).

2 Preliminaries

For an integer n , denote $[n] := \{1, \dots, n\}$. Let $n > k \geq 2$ be positive integers. Let \mathcal{X} be a set of elements of size n . The algorithm does not know \mathcal{X} in advance: it observes the elements in \mathcal{X} one by one according to a uniformly random order. It selects a set of up to k elements $T_{\text{sel}} \subseteq \mathcal{X}$, where an element can only be selected immediately after it is observed, and this decision cannot be reversed.

The value obtained by the algorithm is $f(T_{\text{sel}})$, where $f : 2^{\mathcal{X}} \rightarrow \mathbb{R}_+$ is a set function which is *monotone* and *submodular*. f is monotone if for all $A \subseteq B \subseteq \mathcal{X}$, $f(A) \leq f(B)$. f is submodular if for all $A \subseteq B \subseteq \mathcal{X}$ and all $z \in \mathcal{X} \setminus B$, $f(B \cup \{z\}) - f(B) \leq f(A \cup \{z\}) - f(A)$.

We denote by f_A the marginal submodular function such that for a set B , $f_A(B) := f(B \cup A) - f(A)$. We abuse notation and write $f(z)$ for $z \in \mathcal{X}$ to mean $f(\{z\})$. For any set function f and $A \subseteq \mathcal{X}$, denote $f\{A\} := \{f(z) \mid z \in A\}$. We assume for simplicity that there are no ties in f . OPT is a set that obtains the maximal feasible value of f , $\text{OPT} \in \arg\max_{A \subseteq \mathcal{X}, |A| \leq k} f(A)$. We may assume without loss of generality that $|\text{OPT}| = k$.

In the classical Secretary Problem, the items are also observed in a random order, but only one item is selected, and the goal is to select the maximal-value element with a high probability. Gilbert and Mosteller (1966) show that for any input size n , there is a number $t < n$ such that the optimal strategy is to observe the first t items, set θ to be the maximal value of these items, and then select the first item starting at $t + 1$ whose value is larger than θ . In the limit of $n \rightarrow \infty$, the probability of success of this strategy is $1/e$, and $t/n \rightarrow 1/e$. For finite n, t , the probability of success is $P_{\text{sp}}(n, t) := \frac{t}{n} \sum_{i=t+1}^n \frac{1}{i-1}$, and the optimal t for input size n is $N_{\text{sp}}(n) := \arg\max_{t \in [n]} P_{\text{sp}}(n, t)$. The optimal strategy has a probability of success $P_{\text{sp}}(n) := P_{\text{sp}}(n, N_{\text{sp}}(n))$. Lemma A.1, which we prove in Appendix A, shows that $P_{\text{sp}}(n)$ is monotonic decreasing.

3 Main result

The proposed algorithm, CSS, satisfies the following guarantee.

Theorem 3.1. *Let T_{sel} be the set of elements selected by CSS. Then for $n \geq k \geq 2$,*

$$\mathbb{E}[f(T_{\text{sel}})] \geq 0.1933 \cdot f(\text{OPT}).$$

Moreover, the algorithm uses a memory of order k (up to logarithmic factors in n) and n evaluations of a marginal contribution to f .

For simplicity, we mostly discuss CSS in the case where n is larger than some constant n_0 , and n/k is an integer. In this case, as shown in Section 4 below, CSS splits the input sequence into segments of equal size and selects at most one item in each segment. The competitive ratio lower bound of Theorem 3.1 for $n \geq n_0$ is derived in Section 5, restated as Cor. 5.11. For $n < n_0$, a simple adaptation is required which obtains the same competitive ratio and does not violate the resource constraints. This adaptation is described in Section 6.

Consider now the anytime requirement. Let $t \leq k$ be an integer. As the description of CSS below makes clear, running CSS for some n, k and stopping after $\tau := t \cdot \frac{n}{k}$ items is equivalent to running CSS for input length τ and with cardinality constraint t . Therefore, the competitive ratio guarantee holds also for each prefix of the input sequence: Suppose that the set selected by CSS is examined after τ items. Let OPT_t be the optimal selection of at most t elements out of the first τ items in the input sequence. Then, letting $T_{\text{sel}}[\tau]$ be the elements selected by CSS until item τ has been observed, we have

$$\mathbb{E}[f(T_{\text{sel}}[\tau])] \geq 0.1933 \cdot f(\text{OPT}_t).$$

In the case that $n < n_0$ or n/k is not an integer, the anytime property also holds, with a slight adaptation. We give the exact guarantee for small n in Section 6. A version of the anytime property holds also for the algorithms of Bateni et al. (2010); Feldman et al. (2011), but not for the algorithm of Kesselheim and Tönnis (2016).

4 The CSS algorithm

Alg. 1 lists CSS for the case that $n \geq n_0$ and n/k is an integer. The input sequence is split into k segments of fixed length (as in Bateni et al. 2010). In each segment, CSS attempts to select the element with the maximal marginal improvement. The marginal improvement is evaluated only with respect to *marked* elements, where an element is considered marked only if it was selected by CSS and had the maximal marginal utility in its segment. Denote $\mu_i := \mathbb{I}[\text{segment } i \text{ was marked}]$. We call segments in which a marked element was selected, marked segments. The set of marked elements selected until round i is denoted T_i in CSS. The number of consecutive unmarked segments before round i is denoted r_i . Note that for any $j \in [r_i]$, $T_{i-j} = T_i$. S_i denotes the (unordered) set of elements in segment i . Denote $S_i^j := \cup_{t=i}^j S_t$. For each observed element s in round i , the only evaluation of f performed in

CSS is $f_{T_i}(s)$. Let $\beta \in (0, 1)$ be a constant, and let D be an integer. As will be shown in Cor. 5.11, We achieve our competitive ratio by setting $D = 200$ and $\beta = 0.63$. In CSS, we denote by $\text{Top}_D(i)$ the top D values in $\cup_{j \in [r_i]} f_{T_{i-j}} \{S_{i-j}\} \equiv f_{T_i} \{S_{i-r_i}^{i-1}\}$ and $\text{MaxSeg}(i) := \max f_{T_i} \{S_i\}$. These are implicitly updated by CSS whenever a new element is observed or a new segment starts, using $O(D)$ memory and the values $f_{T_i}(s)$.

Algorithm 1 CSS

input Integers $n \geq k \geq 2$; assume $n \geq n_0$ and n/k is an integer.

```

1:  $T_1 \leftarrow \emptyset, r_1 \leftarrow 0, \mathcal{B} \leftarrow \lceil \beta n/k \rceil$ .
2: for  $i = 1 : k$  do
3:   # Handle segment  $i$  of length  $n/k$ 
4:   if  $r_i = 0$  or  $r_i \geq D$  then {Run the Secretary Problem strategy}
5:     Observe the first  $N_{\text{sp}}(n/k)$  elements, let  $\theta$  be the maximal value of  $f_{T_i}(s)$  for an observed element  $s$ .
6:     Select the first element  $s$  in the rest of the segment such that  $f_{T_i}(s) > \theta$  (if it exists).
7:   else {Run the SwH strategy, see Section 4.1}
8:     Let  $\theta$  be the  $(r_i + 1)$ -largest number in  $\text{Top}_D(i)$ .
9:     Select the first element  $s$  out of the first  $\mathcal{B}$  items in the segment that satisfies  $f_{T_i}(s) > \theta$  (if one exists).
10:    if no element was selected from the first  $\mathcal{B}$  items in the segment then
11:      Let  $\theta'$  be the largest value of  $f_{T_i}(s)$  in the first  $\mathcal{B}$  items
12:      Select out of the rest of the segment the first element  $s$  with  $f_{T_i}(s) > \theta'$  (if one exists).
13:    end if
14:  end if
15:  Let  $s_i$  be the element selected above (or a dummy if no element was selected).
16:  if  $f_{T_i}(s_i) = \text{MaxSeg}(i)$  then {segment  $i$  is marked,  $\mu_i = 1$ }
17:     $r_{i+1} \leftarrow 0, T_{i+1} \leftarrow T_i \cup \{s_i\}$ .
18:  else  $\{\mu_i = 0\}$ 
19:     $r_{i+1} \leftarrow r_i + 1, T_{i+1} \leftarrow T_i$ ,
20:  end if
21: end for

```

When CSS attempts to select an element in a segment, its strategy depends on the number of previous consecutive unmarked segments, r_i . If $r_i = 0$ or $r_i \geq D$, the optimal Secretary Problem strategy is executed in segment i . Otherwise, a different strategy is executed, which uses the information collected in the previous consecutive unmarked segments. This strategy has a higher probability of success than the Secretary Problem strategy, due to the additional information available to it from previous segments. We discuss this strategy further in Section 4.1.

The reason that only information from unmarked segments is used when selecting an element from a segment, is that once a segment is marked, the set T_i is statistically dependent on the items in the previous segments. This means that when marked segments are involved, the distribution of marginal values between segments is not uniformly random. As will be evident in our analysis in Section 5, this uniformity is crucial for guaranteeing a given success probability. We now describe the selection strategy for $r_i \in [D - 1]$ as a solution to a stand-alone selection problem.

4.1 Selecting with history

We define a stand-alone problem of selecting an item, which is a variant of the classical Secretary Problem. We term this problem *Select with History* (SwH). SwH is parametrized by an integer $K \geq 2$, and a finite set of real numbers Z , where $N := |Z|$, such that N/K is a positive integer. In this setting, the numbers in Z are split uniformly at random into two disjoint sets, A_1 and A_2 , such that $|A_2| = N/K$ and $|A_1| = N - N/K$. The numbers in A_2 are then ordered according to a uniformly random permutation σ . In this selection problem, A_1 is observed, and then an item from A_2 is selected. The item is selected as in the secretary problem: $\sigma(A_2)$ is observed in order, and any observed item may only be selected immediately after it is observed, with no possibility of replacing it later.

As we prove in Section 5, this selection problem, when setting $Z = f_{T_{i-r}} \{S_{i-r}^i\}$ and $K = r + 1$ for an integer r , is exactly the problem faced by CSS conditioned on $r_i = r \geq 1$. Denote by $\sigma_{\mathcal{B}}(A_2)$ the first $\mathcal{B} = \lceil \beta N/K \rceil$ items

in A_2 according to the ordering σ . For $r \in [D - 1]$, the selection strategy that CSS implements is equivalent to the following strategy for SwH:

Select, from $\sigma_B(A_2)$, the first element with a value that exceeds the top $r + 1$ values in A_1 . If no item in $\sigma_B(A_2)$ exceeded this threshold, select from the rest of the items in $\sigma(A_2)$ the first element with a value that exceeds $\max \sigma_B(A_2)$.

This strategy is inspired by a strategy given in Gilbert and Mosteller (1966) for the case where the items are drawn i.i.d. from a known distribution. Whereas under a known distribution the first threshold can be set based on this knowledge, here we estimate it based on previous segments. Indeed, it can be shown that in the limit of $K \rightarrow \infty$, the success probability of our strategy is almost the same as that of the strategy proposed in Gilbert and Mosteller (1966).

Consider the probability space defined by the random choices of A_1 and σ . Denote the event that the strategy above succeeds under these random choices by $\text{SwHgood}(Z, K)$. Define, for $K \in \{2, \dots, D\}$,

$$R(N, K) := \mathbb{P}[\text{SwHgood}([N], K)]. \quad (1)$$

Note that the probability of $\text{SwHgood}(Z, K)$ depends only on the size of Z and not on its content, since the strategy only uses relative comparisons. Thus $R(N, K) = \mathbb{P}[\text{SwHgood}(Z, K)]$ for any Z of size N . We also define, for $r = 1$ or $r > D$, $R(N, r) := P_{\text{sp}}(N)$, the probability of succeeding in the Secretary Problem for input length N . We give a formula for R in Section 5 below, and prove that this function gives the conditional probability of success in segment i conditioned on r_i .

5 Analysis

In this section we prove Theorem 3.1. We start by proving, in Section 5.1, Theorem 5.3 which gives a lower bound on the competitive ratio, as a function of the probability of marking each segment. This theorem holds for any algorithm that selects an item from each segment, regardless of the specific selection strategy employed in every round. In Section 5.2 we show that the conditional probability of success of CSS in each segment is given by the probability of success in the SwH problem. In Section 5.3 we show that the sequence of marked/unmarked segments can be analyzed using a Markov chain, and bound the probability of marking each segment using Markov theory. Lastly, in Section 5.4 we analyze the conditional probability of marking a given segment i , conditioned on the number of previous consecutive unmarked segments, r_i . This provides numerical constants which can then be plugged into the theorems of the previous sections, to give the competitive ratio in Cor. 5.11.

5.1 A lower bound on the quality of the solution

To provide a lower bound for the quality of the solution, we first prove a general result that depends on the probability of success in selecting a maximal element in each segment. The following lemma from Bateni et al. (2010) is useful.

Lemma 5.1 (Bateni et al. (2010), Lemma 7). *For a submodular function f , a set R , and a uniformly random subset $A \subseteq R$ of fixed size L , $\mathbb{E}[f(A)] \geq \frac{L}{k} f(R)$.*

Following a construction of Bateni et al. (2010), Define a set P_{k+1} of size k as follows: Let S_i^* be the set of elements from OPT that are in segment i , that is $S_i^* = \text{OPT} \cap S_i$. A representative element is randomly drawn from S_i^* and denoted t_i^* . If $S_i = \emptyset$, then no representative is selected, and t_i^* is set to a dummy element with no contribution to f . Let $P_{j+1} = \{t_1^*, \dots, t_j^*\}$. The value of P_{k+1} can be lower-bounded as follows.

Lemma 5.2. $\mathbb{E}[f(P_{k+1})] \geq (1 - (1 - 1/k)^k) f(\text{OPT})$.

Proof. Let $J = |P_{k+1} \cap \text{OPT}|$. Bateni et al. 2010 (lemma 6) shows that $\mathbb{E}[J] \geq k(1 - (1 - 1/k)^k)$. Since all elements of OPT are equally likely to appear in P_{k+1} , then conditioned on J the set P_{k+1} is uniformly random out of OPT. By Lemma 5.1, it follows that $\mathbb{E}[f(P_{k+1})] \geq \mathbb{E}[\frac{J}{k} f(\text{OPT})] \geq (1 - (1 - 1/k)^k) f(\text{OPT})$. \square

The following theorem quantifies the competitive ratio based on lower bounds on the success probability in each segment.

Theorem 5.3. Let S_1, \dots, S_k be equal-length segments of the input sequence. Let $T_1 = \emptyset$. Suppose that for each $i \leq k$, either $T_{i+1} = T_i$ or $T_{i+1} = T_i \cup \arg\max f_{T_i}(S_i)$. Suppose that $\mathbb{P}[T_{i+1} \neq T_i] \geq p_i$, and assume $p_{i+1} \geq p_i$ for all i . Then

$$\mathbb{E}[f(T_{k+1})] \geq (1 - (1 - 1/k)^k) \frac{\sum_{i=1}^k p_i}{k(1 + p_k)} f(\text{OPT}).$$

Proof. Denote by s_i^{sel} the single element in $T_{i+1} \setminus T_i$, if one exists, or a dummy element with no contribution to f if $T_i = T_{i+1}$. So $T_{i+1} = \{s_1^{\text{sel}}, \dots, s_i^{\text{sel}}\}$. Recall that $P_{i+1} = \{t_1^* \dots t_i^*\}$. Let $P = P_{k+1}$, $T = T_{k+1}$. Denote $p_0 = 0$.

$$\mathbb{E}[f(T)] = \sum_{i=1}^k \mathbb{E}[f_{T_i}(s_i^{\text{sel}})] \geq \sum_{i=1}^k p_i \mathbb{E}[f_{T_i}(t_i^*)] = \sum_{t=1}^k \left((p_t - p_{t-1}) \sum_{i=t}^k \mathbb{E}[f_{T_i}(t_i^*)] \right). \quad (2)$$

We have the following fact (see e.g. Bateni et al. (2010), Lemma 5): $\forall A, B$ sets $\sum_{a \in A} f_B(a) \geq f_B(A)$. Therefore, denoting $P'_t = P \setminus P_t$,

$$\sum_{i=t}^k f_{T_i}(t_i^*) \geq \sum_{i=t}^k f_T(t_i^*) \geq f_T(P'_t) = f(P'_t \cup T) - f(T) \geq f(P'_t) - f(T).$$

Therefore, since $p_t - p_{t-1} \geq 0$,

$$\mathbb{E}[f(T)] \geq \sum_{t=1}^k (p_t - p_{t-1}) \mathbb{E}[f(P'_t)] - \sum_{t=1}^k (p_t - p_{t-1}) \mathbb{E}[f(T)] = \left(\sum_{t=1}^k (p_t - p_{t-1}) \mathbb{E}[f(P'_t)] \right) - p_k \mathbb{E}[f(T)].$$

The ordering in P is uniformly random, therefore P'_t is also a uniformly random set of size $k - t + 1$ from P . By Lemma 5.1, $\mathbb{E}[f(P'_t)] \geq \frac{k-t+1}{k} \mathbb{E}[f(P)]$. Therefore

$$\mathbb{E}[f(T)] \geq \left(\sum_{t=1}^k (p_t - p_{t-1}) \frac{k-t+1}{k} \mathbb{E}[f(P)] \right) - p_k \mathbb{E}[f(T)].$$

Rearranging, we get

$$\mathbb{E}[f(T)] \geq \frac{\sum_{t=1}^k p_t}{k(1 + p_k)} \mathbb{E}[f(P)].$$

Combining this with the lower bound on $f(P)$ in Lemma 5.2, the statement of the theorem follows. \square

The following immediate corollary of Theorem 5.3 gives a single competitive ratio that holds for all values of k , using a finite set of known values p_1, \dots, p_j .

Corollary 5.4. Assume the same conditions as Theorem 5.3. Let $a_t := (1 - (1 - 1/t)^t) \frac{\sum_{i=1}^t p_i}{t(1 + p_t)}$. For any value of k and for any integer j , the elements T_{sel} selected by CSS satisfy

$$\mathbb{E}[f(T_{\text{sel}})] \geq \min \left\{ (1 - 1/e) \frac{\sum_{i=1}^j p_i}{j(1 + p_j)}, \min_{t \in [j]} a_t \right\} f(\text{OPT}).$$

5.2 Conditional probability of success in a segment

Cor. 5.4 gives a lower bound on the competitive ratio of CSS based on lower bounds p_i on the probability that segment i is marked. To derive p_i , we obtain a lower bound for the probability $\mathbb{P}[\mu_i]$. As a first step, we consider the conditional probability of marking segment i , conditioned on the value of r_i , and show, in Theorem 5.5 below, that it is given by the probability of success in the SwH setting. The main challenge is showing that conditioned on r_i , this probability is independent of the identity and ordering of the items in the input sequence before round i .

Unmarked segments do not change the set T that CSS uses to evaluate the marginal contribution of new elements. This is the crucial observation that allows proving this independence claim: The probabilistic process (conditioned on the value of r_i) can be described as first running until iteration $i - r_i$, which determines the set T_{i-r_i} , then deciding how to distribute the items in $S_{i-r_i}^i$ into different segments and how to order each segment, and then running the SwH strategy, where the numbers are the values of the marginal contribution of the items $f_{T_{i-r_i}}(s)$. This holds because all the segments $i - r_i, \dots, i - 1$ are unmarked, and therefore they do not change the set T_i , so that $T_{i-r_i} = T_i$. The argument would thus fail if any of the segments used for the strategy were marked. It is given in detail in the proof of Theorem 5.5 below.

Denote by σ_i the ordering (with respect to some fixed indexing scheme) of the elements in segment i . Denote by S_i the set of items in segment i . Note that S_1, \dots, S_k and $\sigma_1, \dots, \sigma_k$ fully determine the run of the algorithm. Denote the history prior to round i by $V_i = (S_1, \dots, S_{i-1}), (\sigma_1, \dots, \sigma_{i-1})$. We show that the ordering and identity of items until round $i - r_i$ have no effect on the probability of success in round i and that this probability is given by the function R defined in Eq. (1).

Theorem 5.5. *Let $n \geq k \geq 2$ such that n/k is an integer. For any integer $r \in \{0, \dots, k - 1\}$ and any round i ,*

$$\mathbb{P}[\mu_i = 1 \mid r_i = r, S_{i-r}^i, V_{i-r}] = \mathbb{P}[\mu_i = 1 \mid r_i = r] = R(n(r+1)/k, r+1). \quad (3)$$

To prove Theorem 5.5 we use an auxiliary property and an induction argument. For a given r , denote by $\text{uprob}(r)$ the property that Eq. (3) holds for all rounds i . In addition, define a conditional independence property $\text{indep}(r)$ as follows

$$\text{indep}(r) := \forall j \in \{r+1, \dots, k\}, \quad (r_{j+1} = r+1) \perp S_{j-r}^j \mid (r_{j-r} = 0), V_{j-r}.$$

Here $X \perp Y \mid Z$ denotes the statistical independence of X and Y conditioned on Z . To prove Theorem 5.5 we must show that $\forall r \in \{0, \dots, k - 1\}$, $\text{uprob}(r)$ holds. We prove this theorem by proving the following three claims:

Claim 1 For $r = 0$ or $r \geq D$, $\text{indep}(r)$ and $\text{uprob}(r)$ hold;

Claim 2 For $r \in [D - 1]$, if $\text{indep}(r - 1)$ holds then $\text{uprob}(r)$ holds;

Claim 3 For all $r \in [D - 1]$, $\text{indep}(r - 1)$ and $\text{uprob}(r)$ implies $\text{indep}(r)$.

These three claims immediately imply Theorem 5.5. We now prove each of these claims.

Proof of Claim 1. In round i , if $r_i = 0$ or $r_i \geq D$, then CSS runs the optimal secretary problem strategy for input size n/k on the values $f_{T_i}\{S_i\}$. The probability of success of this strategy depends only on the rank ordering of these values. This rank ordering is uniformly random conditioned on T_i , which is determined by V_i . Therefore the probability of success is the same as the probability of success of this strategy in the original Secretary Problem, $P_{\text{sp}}(n/k) \equiv R(n/k, 1)$. Since conditioned on $r_i = 0$, we have $r_{i+1} = \mu_i$, this further implies $\text{indep}(0)$. \square

Proof of Claim 2. We show that for $r \in [D - 1]$, $(\mu_i \mid r_i = r, S_{i-r}^i, V_{i-r})$ is distributed like $\text{SwHgood}(f_{T_{i-r}}\{S_{i-r}^i\}, r+1)$. First, note that conditioned on $r_i = r, T_{i-r}, S_{i-r}^i$, the success of the $\text{SwH}(f_{T_{i-r}}\{S_{i-r}^i\}, r+1)$ strategy depends only on the identity of items from S_{i-r}^i that are in S_i and on their ordering σ_i . From the assumption that $\text{indep}(r - 1)$ holds (setting $j = i - 1$), we have that $S_{i-r}^{i-1} \perp (r_i = r) \mid r_{i-r} = 0, V_{i-r}$. It follows that $S_{i-r}^{i-1} \mid r_i = r, V_{i-r}$ is uniformly random out of S_{i-r}^k (since $r_i = r$ implies $r_{i-r} = 0$). Since also S_i is uniformly random out of S_i^k conditioned on $r_i = r, V_{i-r}, S_{i-r}^{i-1}$, it follows that $S_i \mid r_i = r, V_{i-r}, S_{i-r}^i$ is a uniformly random set out of S_{i-r}^i .

Since T_{i-r} is a function of V_{i-r} , S_i is uniformly random also conditioned on $r_i = r, V_{i-r}, S_{i-r}^i, T_{i-r}$. It follows that $f_{T_{i-r}}\{S_i\}$ is a uniformly random subset of size n/k out of the set $f_{T_{i-r}}\{S_{i-r}^i\}$ of size $n(r+1)/k$. The ordering σ_i of S_i is also clearly uniformly random conditioned on V_i . Therefore $f_{T_{i-r}}\{S_i\}, \sigma_i$ are distributed like A_2, σ in $\text{SwHgood}(f_{T_{i-r}}\{S_{i-r}^i\}, r+1)$.

Conditioned on $r_i = r$, we have that for all $j \in [r]$, $T_i = T_{i-j}$. Therefore $\cup_{j \in [r]} f_{T_{i-j}}\{S_{i-j}\} = f_{T_{i-r}}\{S_{i-r}^i\}$, which implies that $\text{Top}_D(i)$ is the set of top D values in $f_{T_{i-r}}\{S_{i-r}^i\}$. Therefore, the threshold θ is exactly the top $r+1$ value in $f_{T_{i-r}}\{S_{i-r}^i\}$. It follows that CSS implements the strategy defined in Section 4.1 on the set $f_{T_{i-r}}\{S_{i-r}^i\}$. Therefore the probability of $\mu_i = 1$ under the above conditions is exactly the probability of the event $\text{SwHgood}(f_{T_{i-r}}\{S_{i-r}^i\}, r+1)$. \square

Proof of Claim 3. From $\text{indep}(r-1)$, using $j = i-1$, we have $(r_i = r) \perp S_{i-r}^{i-1} \mid r_{i-r} = 0, V_{i-r}$. It also holds that $(r_i = r) \perp S_i \mid S_{i-r}^{i-1}, r_{i-r} = 0, V_{i-r}$, since S_i is observed after r_i is set. Therefore it follows that

$$(r_i = r) \perp S_{i-r}^i \mid r_{i-r} = 0, V_{i-r}. \quad (4)$$

From $\text{uprob}(r)$ we have $(\mu_{i+1} = 1) \perp S_{i-r}^i \mid r_i = r, V_{i-r}$. Since $r_{i+1} = r_i + \mu_i$, it follows that

$$(r_{i+1} = r+1) \perp S_{i-r}^i \mid r_i = r, V_{i-r}.$$

Since $r_i = r$ implies $r_{i-r} = 0$, it follows that

$$(r_{i+1} = r+1) \perp S_{i-r}^i \mid r_i = r \wedge r_{i-r} = 0, V_{i-r}. \quad (5)$$

From Eq. (4) and Eq. (5), using the contraction principle for conditional independence (Pearl, 2003),

$$(r_{i+1} = r+1) \perp S_{i-r}^i \mid r_i = 0, V_{i-r}.$$

□

Having proved the three claims, this proves Theorem 5.5.

5.3 Unconditional probability of success in a segment

We now analyze the unconditional probability of marking a segment, using the conditional probabilities given in Theorem 5.5. From Theorem 5.5 we have (since V_i determines r_1, \dots, r_{i-1})

$$\mathbb{P}[\mu_i = 1 \mid r_i = r, r_1, \dots, r_{i-1}] = R(n(r+1)/k, r+1).$$

Conditioned on $r_i = r$, the value of μ_i determines r_{i+1} . Therefore the random sequence r_1, r_2, \dots, r_k is a Markov chain. We define a general form of a Markov chain such that this sequence is generated by a chain of this form. Let $\text{MC}(M, k)$ be a Markov chain on the state space $\{0, 1, \dots, k-1\}$, parametrized by a function $M : \{0, \dots, k-1\} \rightarrow (0, 1)$, with a starting state 0, and transition probabilities given by

$$P(r', r) := \mathbb{P}[r_i = r \mid r_{i-1} = r'] = \begin{cases} M(r') & r = 0 \\ 1 - M(r') & r = r' + 1 \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Denote the corresponding transition matrix by P . This Markov chain is a variation of the “Winning streak” chain (Levin et al., 2009, example 4.15). Denote the distribution of states at iteration i by π_i . We have $\mathbb{P}[\mu_i = 1] = \mathbb{P}[r_{i+1} = 0] = \pi_i(0)$. In the following lemma, we lower-bound $\pi_i(0)$ by relating it to the stationary distribution π of this Markov chain, and by further bounding $\pi(0)$. This lower bound will then be used to set p_i in Cor. 5.4, using the substitution $M(r) := R(n(r+1)/k, r+1)$.

Theorem 5.6. *Suppose that for all $i \geq 0$, $M(i) \geq \alpha$. Let π_i be the distribution of states at iteration i of the Markov chain $\text{MC}(M, k)$. Then for all $i \geq j$,*

$$\mathbb{P}[\mu_i = 1] = \pi_i(0) \geq \left(\sum_{t=0}^j \prod_{l=0}^{t-1} (1 - M(l)) + (1 - \alpha)^{j+1} / \alpha \right)^{-1} - (1 - \alpha)^j.$$

In addition, for all i , $\pi_i(0) = e_0^T P[i]^i e_0$, where $P[i]$ is the sub-matrix of P with rows and columns $\{0, 1, \dots, i\}$, and e_0 is the unit vector $(1, 0, \dots, 0)$ of size $i+1$.

To prove the theorem, we consider first the stationary distribution π of the Markov chain defined above. This Markov chain is clearly *irreducible* (see, e.g. Levin et al., 2009), that is: there is a path with probability greater than zero between any two states in the chain. By (Levin et al., 2009, Proposition 1.14), if the Markov chain is irreducible, then there exists a stationary distribution π , and $\pi(0) = 1/\mathbb{E}[\tau_0]$, where $\tau_0 := \min\{t \geq 1 \mid r_{t+1} = 0\}$ is the number of steps until the chain returns to state 0 for the first time. We have

$$\begin{aligned}\mathbb{E}[\tau_0] &= \sum_{t=0}^{\infty} \mathbb{P}[\tau_0 > t] = \sum_{t=0}^{\infty} \prod_{j=0}^{t-1} (1 - M(j)) \\ &= \sum_{t=0}^T \prod_{j=0}^{t-1} (1 - M(j)) + \sum_{t=T+1}^{\infty} \prod_{j=1}^t (1 - M(j)) \\ &\leq \sum_{t=0}^T \prod_{j=0}^{t-1} (1 - M(j)) + \sum_{t=T+1}^{\infty} (1 - \alpha)^t \\ &\leq \sum_{t=0}^T \prod_{j=0}^{t-1} (1 - M(j)) + \frac{(1 - \alpha)^{T+1}}{\alpha}.\end{aligned}$$

Therefore

$$\pi(0) \geq \left(\sum_{t=0}^j \prod_{l=0}^{t-1} (1 - M(l)) + (1 - \alpha)^{j+1}/\alpha \right)^{-1}. \quad (7)$$

In order to bound $\pi_i(0)$ using this bound on $\pi(0)$, the following lemma gives a lower bound on distance between the two quantities.

Lemma 5.7. *If $M(r) \geq \alpha$ for all $r \geq 0$, then for all integers t , $|\pi_t(0) - \pi(0)| \leq (1 - \alpha)^t$.*

Proof. Let $(X_t, Y_t)_{t=0}^{\infty}$ be a joint random process (a coupling) of two Markov chains X, Y such that both chains follow the transition matrix P . Define the meeting time of the two chains by $\tau := \min\{t : X_t = Y_t\}$.

By Levin et al. 2009, (Corollary 5.3 and Definition 4.22),

$$|\pi_t(0) - \pi(0)| \leq \max_{x,y} \mathbb{P}[\tau > t \mid X_0 = x, Y_0 = y]. \quad (8)$$

We define the coupling below such that for all X_i ,

$$\mathbb{P}[Y_{i+1} = 0 \mid Y_i, X_i] = P(Y_i, 0). \quad (9)$$

In addition, we set $Y_{i+1} = Y_i + 1$ whenever $Y_{i+1} \neq 0$. This guarantees that the chain Y follows the transition matrix P , that is $\mathbb{P}[Y_{i+1} = y \mid Y_i = y'] = P(y', y)$ for all states y, y' . We make sure the same properties symmetrically hold for X , thus the same guarantee also holds for X . Given X_i, Y_i , the next states X_{i+1}, Y_{i+1} are set as follows: select the chain $Z \in \{X, Y\}$ with the smallest $P(Z_i, 0)$. First, suppose that $Z = X$. In this case draw $X_{i+1} \in \{0, X_i + 1\}$ according to an independent coin toss such that $\mathbb{P}[X_{i+1} = 0 \mid X_i, Y_i] = P(X_i, 0)$. Afterwards, if $X_{i+1} = 0$ set $Y_{i+1} = 0$ deterministically. If $X_{i+1} \neq 0$, draw $Y_{i+1} \in \{0, Y_i + 1\}$ using a different independent coin, such that

$$\mathbb{P}[Y_{i+1} = 0 \mid Y_i, X_i, X_{i+1} \neq 0] = \frac{P(Y_i, 0) - P(X_i, 0)}{1 - P(X_i, 0)}.$$

Under this process we have $\mathbb{P}[X_{i+1} = 0 \mid Y_i, X_i] = P(X_i, 0)$, and

$$\mathbb{P}[Y_{i+1} = 0 \mid Y_i, X_i] = 1 \cdot P(X_i, 0) + \frac{P(Y_i, 0) - P(X_i, 0)}{1 - P(X_i, 0)} \cdot (1 - P(X_i, 0)) = P(Y_i, 0).$$

Thus Eq. (9) holds for both X and Y . If $Z = Y$ an equivalent symmetric process takes place, again satisfying Eq. (9).

We now use Eq. (8), by providing an upper bound for $\mathbb{P}_{x,y}[\tau > t]$, where $\mathbb{P}_{x,y}$ stands for conditioning on $X_0 = x, Y_0 = y$. We have $\tau \leq \min\{t \mid X_t = Y_t = 0\}$. Define the event $E_i = ((X_i \neq 0) \vee (Y_i \neq 0))$. We have

$$\mathbb{P}_{x,y}[\tau > t] \leq \mathbb{P}_{x,y}[\bigwedge_{i=1}^t E_i] = \prod_{i=1}^t \mathbb{P}[E_i \mid E_1, \dots, E_{i-1}]. \quad (10)$$

We have

$$\mathbb{P}[E_i \mid E_1, \dots, E_{i-1}] = \mathbb{E}_{X_{i-1}, Y_{i-1}}[\mathbb{P}[E_i \mid X_{i-1}, Y_{i-1}] \mid E_1, \dots, E_{i-1}],$$

Therefore it suffices to upper bound $\mathbb{P}[E_i \mid X_{i-1}, Y_{i-1}]$. Fix some $X_{i-1} = x_{i-1}, Y_{i-1} = y_{i-1}$, and assume without loss of generality $P(x_{i-1}, 0) < P(y_{i-1}, 0)$. Then

$$\begin{aligned} \mathbb{P}[\neg E_i \mid X_{i-1} = x_{i-1}, Y_{i-1} = y_{i-1}] \\ &= \mathbb{P}[X_i = 0 \mid X_{i-1} = x_{i-1}, Y_{i-1} = y_{i-1}] \cdot \mathbb{P}[Y_i = 0 \mid X_i = 0, X_{i-1} = x_{i-1}, Y_{i-1} = y_{i-1}] \\ &= P(x_{i-1}, 0) \cdot 1 = M(x_{i-1}) \geq \alpha. \end{aligned}$$

Plugging this into Eq. (10) we get $\mathbb{P}_{x,y}[\tau > t] \leq (1 - \alpha)^t$. Therefore, by Eq. (8), the statement of the lemma holds. \square

The proof of Theorem 5.6 is now immediate: The first part of Theorem 5.6 directly follows from Eq. (7) and Lemma 5.7. The second part of the theorem follows since in this Markov chain, it is impossible to reach states larger than i in the first i steps. Therefore $\pi_i(0) = e_0^T P[i] e_0$.

5.4 A lower bound on the competitive ratio of CSS

In this section we provide formulas for R and calculate its limit when $N \rightarrow \infty$. We use this along with Theorem 5.6 to infer values for p_i , which lower bound the unconditional probability of marking a segment. We then plug these values into Cor. 5.4 to conclude the competitive ratio of CSS for the case of $n \geq n_0$ and n/k an integer. For an integer $K \geq 1$, define

$$Q(K) := \lim_{L \rightarrow \infty} R(LK, K).$$

For $K = 1$ or $K > D$, we have from the definition of $R(N, r)$ that $Q(K) = 1/e$. For $K \in \{2, \dots, D\}$, the following lemma gives the value of $Q(K)$.

Lemma 5.8. *Let $K \in \{2, \dots, D\}$. Then*

$$Q(K) = \beta \log(1/\beta) \left(1 - \frac{1}{K}\right)^K + \sum_{j=1}^{\infty} \binom{j+K-1}{K-1} \left(1 - \frac{1}{K}\right)^K \frac{1}{K^j} \left(\frac{1 - (1-\beta)^j}{j} + \beta \int_{\beta}^1 \frac{(1-x)^{j-1}}{x} dx \right). \quad (11)$$

Proof. Let $N = LK$. We calculate $R(N, K)$ based on the definition in Eq. (1), and then take the limit. Let $Z = \{z_1, \dots, z_N\}$ where $z_i > z_{i+1}$ for all i . Denote $G := \text{SwHgood}(Z, K)$. Let θ be the K 'th largest number in A_1 , and let θ' be the largest number in $\sigma_{\mathcal{B}}(A_2)$, where $\mathcal{B} = \lceil \beta L \rceil$. The strategy described in Section 4.1 selects the first element observed in A_2 which is larger than θ if one is found in the first \mathcal{B} items in A_2 . Otherwise, it selects the first element in the rest of the items that is larger than θ' . Let $J = |\{z \in A_2 \mid z \geq \theta\}|$. We have

$$\mathbb{P}[G] = \sum_{j=0}^L \mathbb{P}[G \mid J = j] \mathbb{P}[J = j]. \quad (12)$$

We calculate the two terms in the multiplication. To calculate $\mathbb{P}[J = j]$, let I be the index such that $\theta = z_I$. We have $J = I - K$. If $I = i$, this means that out of the numbers z_1, \dots, z_{i-1} , exactly $K - 1$ are in A_1 , and also $z_i \in A_1$. Since $|A_1| = N - L$ and its content is allocated uniformly at random, we have

$$\mathbb{P}[J = i - K] = \mathbb{P}[I = i] = \binom{i-1}{K-1} \prod_{l=0}^{K-1} \frac{N-L-l}{N-l} \prod_{l=0}^{i-K-1} \frac{L-l}{N-K-l}.$$

Therefore

$$\mathbb{P}[J = j] = \binom{j+K-1}{K-1} \prod_{l=0}^{K-1} \frac{N-L-l}{N-l} \prod_{l=0}^{j-1} \frac{L-l}{N-K-l}. \quad (13)$$

Taking the limit for $L \rightarrow \infty$ (recalling $N = LK$) we get

$$\lim_{L \rightarrow \infty} \mathbb{P}[J = j] = \binom{j+K-1}{K-1} \left(1 - \frac{1}{K}\right)^K \frac{1}{K^j}. \quad (14)$$

We now calculate $\mathbb{P}[G \mid J = j]$. If $j = 0$, then all $z \in A_2$ have $z < \theta$, therefore no element will be selected from the first \mathcal{B} elements of A_2 . The probability of success is thus exactly as the probability of success of the Secretary Problem strategy with threshold \mathcal{B} , hence $P_{\text{sp}}(L, \mathcal{B})$. We have, following the analysis in Ferguson (1989),

$$\lim_{L \rightarrow \infty} P_{\text{sp}}(L, \mathcal{B}) = \lim_{L \rightarrow \infty} \frac{[\beta L]}{L} \sum_{i=\mathcal{B}+1}^L \frac{1}{i-1} = \lim_{L \rightarrow \infty} \frac{[\beta L]}{L} \sum_{i=\mathcal{B}+1}^L \frac{1}{L} \left(\frac{L}{i-1}\right) = \beta \int_{\beta}^1 \frac{1}{x} dx = \beta \log(1/\beta).$$

Hence

$$\lim_{L \rightarrow \infty} \mathbb{P}[G \mid J = 0] = \beta \log(1/\beta). \quad (15)$$

If $j > 0$, let I be the index in A_2 of the element with a maximal value. Note that if the strategy does not select anything before reaching item I , it will certainly select item I since it is larger than both θ and θ' . Distinguish two cases:

1. If $I = i \leq \mathcal{B}$, then item i is selected as long as all other $j-1$ items that exceed θ are located after item i . Hence, for $i \leq \min(\mathcal{B}, L-j+1)$

$$\mathbb{P}[G \mid I = i \leq \mathcal{B}, J = j] = \prod_{l=0}^{j-2} \frac{L-i-l}{L-1-l}.$$

2. If $I = i > \mathcal{B}$, then item i is selected as long as all other $j-1$ items that exceed θ are located after item i , and also the maximal item in the first $i-1$ items is in the first \mathcal{B} items, so that item i is the first item that is larger than θ' . Hence, for $\mathcal{B} \leq i \leq L-j+1$,

$$\mathbb{P}[G \mid I = i \leq \mathcal{B}, J = j] = \prod_{l=0}^{j-2} \frac{L-i-l}{L-1-l} \frac{\mathcal{B}}{i-1}.$$

Therefore, for $j \geq 1$,

$$\mathbb{P}[G \mid J = j] = \frac{1}{L} \sum_{i=1}^{\min(\mathcal{B}, L-j+1)} \prod_{l=0}^{j-2} \frac{L-i-l}{L-1-l} + \frac{1}{L} \sum_{i=\mathcal{B}+1}^{L-j+1} \prod_{l=0}^{j-2} \frac{L-i-l}{L-1-l} \frac{\mathcal{B}}{i-1}.$$

For any fixed j and $\epsilon \in (0, 1)$, take L large enough such that $j < \epsilon L$. Then, for $j \geq 1$,

$$\begin{aligned} \mathbb{P}[G \mid J = j] &\geq \frac{1}{L} \sum_{i=1}^{\mathcal{B}} \prod_{l=0}^{j-2} \frac{L(1-\epsilon)-i}{L-1} + \frac{1}{L} \sum_{i=\mathcal{B}+1}^{(1-\epsilon)L} \prod_{l=0}^{j-2} \frac{L(1-\epsilon)-i}{L-1} \frac{\mathcal{B}}{i-1} \\ &= \sum_{i=1}^{[\beta L]} \frac{1}{L} \prod_{l=0}^{j-2} \left(1 - \epsilon - \frac{i}{L-1}\right) + \frac{[\beta L]}{L} \sum_{i=1}^{\mathcal{B}} \frac{1}{L} \prod_{l=0}^{j-2} \left(1 - \epsilon - \frac{i}{L-1}\right) \frac{L}{i-1}. \end{aligned}$$

Taking the limit $L \rightarrow \infty$ and $\epsilon \rightarrow 0$, this gives, for $j \geq 1$,

$$\lim_{L \rightarrow \infty} \mathbb{P}[G \mid J = j] \geq \int_0^{\beta} (1-x)^{j-1} dx + \beta \int_{\beta}^1 \frac{(1-x)^{j-1}}{x} dx = \frac{1 - (1-\beta)^j}{j} + \beta \int_{\beta}^1 \frac{(1-x)^{j-1}}{x} dx.$$

Combining Eq. (12), Eq. (14), Eq. (15) and the inequality above, we get that for any fixed j_0 , for a large enough L

$$\mathbb{P}[G] \geq \beta \log(1/\beta) \left(1 - \frac{1}{K}\right)^K + \sum_{j=1}^{j_0} \binom{j+K-1}{K-1} \left(1 - \frac{1}{K}\right)^K \frac{1}{K^j} \left(\frac{1 - (1-\beta)^j}{j} + \beta \int_{\beta}^1 \frac{(1-x)^{j-1}}{x} dx \right).$$

By Eq. (12) we also have $\lim_{L \rightarrow \infty} \mathbb{P}[G] \leq \sum_{j=0}^{\infty} \lim_{L \rightarrow \infty} \mathbb{P}[G \mid J = j] \mathbb{P}[J = j]$ which gives the equality Eq. (11). \square

Based on numeric optimization, we set $\beta = 0.63$. Using numeric calculation, it can be found that the first few values of Q , up to 6 decimal points, with this setting of β are:

$$(Q(1), \dots, Q(10)) = (0.367879, 0.474069, 0.514016, 0.528909, 0.536646, \\ 0.541375, 0.544561, 0.546852, 0.548579, 0.549926).$$

To calculate the competitive ratio of CSS, we further use the following easy lower bound. For $\beta = 0.63$, this lower bound is larger than 0.303.

Lemma 5.9. *There is some n_0 such that for integers $N \geq n_0$, $K \geq 1$ such that N/K is an integer, $R(N, K) \geq \beta(\log(1/\beta)/2 + 1/4)$.*

Proof. By Eq. (11) (taking only $j = 0$ and $j = 1$ from the sum), for $K \in \{2, \dots, D\}$ and a large enough N ,

$$R(N, K) \geq \beta \log(1/\beta) \left(1 - \frac{1}{K}\right)^K + \left(1 - \frac{1}{K}\right)^K (\beta + \beta \log(1/\beta)) \geq \beta(\log(1/\beta)/2 + 1/4).$$

The last inequality follows since $(1 - 1/K)^K \geq 1/4$. For $K = 1$ or $K > D$, we have $R(N, 1) = P_{\text{sp}}(N)$. By Lemma A.1, $P_{\text{sp}}(N) \geq \lim_{t \rightarrow \infty} P_{\text{sp}}(t) = 1/e$. It is easy to check that for any β , $1/e$ is larger than the lower bound above. \square

To obtain our competitive ratio lower bound, we now find numeric lower bounds for $\mathbb{P}[\mu_i = 1]$.

Theorem 5.10. *Assume CSS runs with $D = 200$ and $\beta = 0.63$. There is some n_0 such that for all $n \geq n_0$ and for all k such that n/k is an integer, the following values for p_i satisfy $p_i \geq \mathbb{P}[\mu_i = 1]$, for all integers i . Let $\phi := 0.441086$. For every $i \geq 10$, set $p_i := \phi$. For $i < 10$, set:*

$$(p_1, \dots, p_9) = (0.367879, 0.435004, 0.441157, 0.440969, 0.441042, \\ 0.441074, 0.441082, 0.441084, 0.441085). \quad (16)$$

Proof. By Theorem 5.5, the Markov chain $\text{MC}(M, k)$ with $M(r) := R(n(r+1)/k, r+1)$ describes the transition probabilities of r_1, \dots, r_k . By the definition of Q , for any $\epsilon > 0$ there is a large enough n such that $M(r) \geq Q(r+1) - \epsilon$. We use Theorem 5.6 with M as defined above and $\alpha = 0.303$ (by Lemma 5.9) to obtain that for any $i \geq j$,

$$\mathbb{P}[\mu_i = 1] \geq \pi_i(0) \geq \left(\sum_{t=0}^j \prod_{l=1}^t (1 - Q(l) + \epsilon) + (1 - \alpha)^{j+1} / \alpha \right)^{-1} - (1 - \alpha)^j.$$

Setting j to 200 and numerically calculating $Q(l)$ using Eq. (11), we get that for any $i \geq 200$, and for large enough n , $\mathbb{P}[\mu_i = 1] \geq 0.441086$. For $i < 200$, we use the equality $\mathbb{P}[\mu_i = 1] = e_0^T P[i] e_0$ from Theorem 5.6, again calculating these values numerically. We get the lower bounds in Eq. (16) for $\pi_1(0), \dots, \pi_9(0)$, and further $\pi_i(0) \geq 0.441086$ for $i \in \{10, \dots, 200\}$. \square

The final corollary gives the competitive ratio of CSS, thus proving Theorem 3.1.

Corollary 5.11. *Assume CSS runs with $D = 200$ and $\beta = 0.63$. There is some n_0 such that for all $n \geq n_0$ and for all k such that n/k is an integer, the elements T_{sel} selected by our algorithm satisfy*

$$\mathbb{E}[f(T_{\text{sel}})] \geq 0.1933 \cdot f(\text{OPT}).$$

Proof. We apply Cor. 5.4 as follows: Let a_k be defined as in Cor. 5.4. Set $j := 200$. By Theorem 5.10, we can set $p_i = \phi$ for $i > 9$, and p_i as in Eq. (16) for $i \leq 9$. By calculating a_k numerically for $k \geq j$, we get $\min_{k \in [j]} a_k \geq 0.1935$. In addition, we get numerically that $(1 - 1/e) \frac{\sum_{i=1}^j p_i}{j(1+p_j)} \geq 0.1933$. By Cor. 5.4 this proves the claim. \square

6 Adapting CSS to small n

We have shown guarantees for CSS for $n \geq n_0$ where n/k is an integer. If this is not the case, CSS can be adapted so that the guarantees still hold and the resource constraints are not violated. Clearly, the competitive ratio would remain the same if dummy elements that do not contribute to f were added to the input sequence at random locations. An equivalent adaptation, which does not require additional memory, can be implemented by making the following changes to CSS:

The input sequence of size n should be split into segments with a length determined via a multinomial distribution with n trials and a probability of $1/k$ in each segment². This is equivalent to splitting an input sequence padded with dummies into equal segments, when the padded input sequence is infinitely long. The length of segment i can be set after the end of segment $i - 1$ to $n_i \sim \text{Binomial}(n - \sum_{j=1}^{i-1} n_j, 1/(k - i + 1))$.

There are several ways to emulate the selection strategy in each segment so that it is equivalent to selecting from an infinite sequence padded with dummy elements. For segments with $r_i = 0$, this can be achieved by setting the number of elements that determine the threshold according to $\text{Binomial}(n_i, 1/e)$. For segments with $r_i > 0$, the strategy should take into account the case that $\sum_{j=i-r_i}^{i-1} n_j < r_i$, and in this case set $\theta = 0$.

This approach guarantees that the competitive ratio remain the same although n is small, and without requiring more memory or more evaluations of f .

It is less immediate that the anytime property holds here, since in this case the input sequence is not split into equal length segments. Therefore, when examining the algorithm's selected set after $\tau = tn/k$ items have been observed, it is possible that the number of completed segments by this time is less than t . Nonetheless, the following theorem shows that for large k the anytime property does approximately hold, except for a vanishing fraction of the stream.

Theorem 6.1. *Let $k \geq 2$. Let $\tau < n$ be an integer, and let $t = \tau k/n$. Let $T_{\text{sel}}[\tau]$ be the elements selected by CSS until item τ is observed. Let OPT_t be the optimal selection of at most t elements out of the first τ items in the input sequence. Then, for $t/k = \tilde{\omega}(1/\sqrt{n})$,*

$$\mathbb{E}[f(T_{\text{sel}}[t])] \geq 0.1933(1 - o(1))f(\text{OPT}_t).$$

Proof. For integers N, j , denote by X_N the first N elements of the input sequence, and denote by $\text{OPT}_{N,j}$ the set maximizing $f(S)$, over sets $S \subseteq X_N$ of size at most j . Let j be the largest index such that $N_j := \sum_{i=1}^j n_i \leq \tau$. Let $V = \text{OPT}_{\tau,t} \cap X_{N_j}$. Conditioned on the size of V , the contents of V are uniformly random from $\text{OPT}_{\tau,t}$. Therefore, by Lemma 5.1, $\mathbb{E}[f(V)] \geq \mathbb{E}[\frac{|V|}{t} f(\text{OPT}_{\tau,t})]$. Let V' be a random subset of V of size j if $|V| \geq j$. Otherwise, let $V' = V$. Then $V' \subseteq X_{N_j}$, therefore, applying Lemma 5.1 again,

$$\mathbb{E}[f(\text{OPT}_{N_j,j})] \geq \mathbb{E}[f(V')] \geq \mathbb{E}[\frac{j}{|V|} f(V)] \geq \mathbb{E}[\frac{j}{t} f(\text{OPT}_{\tau,t})].$$

Let $l = n/k$. Now, $\mathbb{E}[j] = \mathbb{E}[N_j]/l = (\tau - \mathbb{E}[\tau - N_j])/l = t - \mathbb{E}[\tau - N_j]/l$. Fix i . Note that $\mathbb{E}[N_{t-i}] = (t - i)l$. Further, if $N_{t-i} \leq \tau$ then $\tau - N_j \leq \tau - N_{t-i}$. Define the event

$$E := (|N_{t-i} - (t - i)l| \leq il) \equiv (|N_{t-i}/n - (t - i)/k| \leq i/k).$$

Under this event, $N_{t-i} \leq \tau$ and $\tau - N_{t-i} \leq 2il$. Therefore, $\mathbb{E}[\tau - N_j] \leq \mathbb{P}[E] \cdot 2il + \mathbb{P}[\neg E]n$. By Hoeffding's inequality, $\mathbb{P}[\neg E] \leq 2 \exp(-2ni^2/k^2)$, hence $\mathbb{E}[\tau - N_j] \leq 2il + 2n \exp(-2ni^2/k^2)$. Therefore

$$\mathbb{E}[j] \geq t - 2i - 2k \exp(-2ni^2/k^2).$$

We obtain,

$$\mathbb{E}[f(\text{OPT}_{N_j,j})] \geq (1 - \frac{2i + 2k \exp(-2ni^2/k^2)}{t})f(\text{OPT}_{\tau,t}).$$

Setting $i = k\sqrt{\log(2n)/2n}$, we get

$$\mathbb{E}[f(\text{OPT}_{N_j,j})] \geq (1 - \frac{k(\sqrt{\log(2n)/2n} + 1/n)}{t})f(\text{OPT}_{\tau,t}) = (1 - \frac{k \cdot \tilde{o}(1/\sqrt{n})}{t})f(\text{OPT}_{\tau,t}).$$

²This is the segment splitting strategy in Feldman et al. (2011)

Applying Theorem 3.1 to X_{N_j} , we have $\mathbb{E}[f(T_{j+1})] \geq 0.1933 \cdot \mathbb{E}[f(\text{OPT}_{N_j,j})]$. This implies the statement of the theorem. \square

7 Conclusion

CSS is an anytime algorithm for the Submodular Secretary Problem under a cardinality constraint. It uses only one evaluation of f per item, and a memory of order k , and improves the competitive ratio compared to the state of the art under these constraints. Order k memory is required in CSS only for storing T_i so that f_{T_i} can be evaluated on new items. If the value of the marginal is provided by an external oracle, a reasonable scenario if f is measured and not calculated, then CSS requires a memory of only $O(\log(n/k))$. Our analysis uses new tools for analyzing algorithms with a variable success rate in different segments. We believe these constructions will be useful also for other online selection problems, and plan to study those in future work.

References

- M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. A knapsack secretary problem with applications. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques*, pages 16–28. Springer, 2007a.
- M. Babaioff, N. Immorlica, and R. Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 434–443. Society for Industrial and Applied Mathematics, 2007b.
- M. Bateni, M. Hajiaghayi, and M. Zadimoghaddam. Submodular secretary problem and extensions. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 39–52. Springer, 2010.
- M. Bateni, M. Hajiaghayi, and M. Zadimoghaddam. Submodular secretary problem and extensions. *ACM Transactions on Algorithms (TALG)*, 9(4):32, 2013.
- G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.
- T. L. Dean and M. S. Boddy. An analysis of time-dependent planning. In *AAAI*, volume 88, pages 49–54, 1988.
- N. B. Dimitrov and C. G. Plaxton. Competitive weighted matching in transversal matroids. In *International Colloquium on Automata, Languages, and Programming*, pages 397–408. Springer, 2008.
- E. B. Dynkin. The optimum choice of the instant for stopping a markov process. In *Sov. Math. Dokl*, volume 4(52), pages 627–629, 1963.
- M. Feldman and R. Zenklusen. The submodular secretary problem goes linear. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 486–505. IEEE, 2015.
- M. Feldman, J. S. Naor, and R. Schwartz. Improved competitive ratios for submodular secretary problems. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 218–229, 2011.
- M. Feldman, O. Svensson, and R. Zenklusen. A simple $o(\log \log(\text{rank}))$ -competitive algorithm for the matroid secretary problem. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1189–1201. Society for Industrial and Applied Mathematics, 2015.
- T. S. Ferguson. Who solved the secretary problem? *Statistical science*, pages 282–289, 1989.
- J. P. Gilbert and F. Mosteller. Recognizing the maximum of a sequence. *Journal of the American Statistical Association*, 61(313):35–73, 1966.

- A. Gupta, A. Roth, G. Schoenebeck, and K. Talwar. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *International Workshop on Internet and Network Economics*, pages 246–257. Springer, 2010.
- T. Kesselheim and A. Tönnis. Submodular secretary problems: Cardinality, matching, and linear constraints. *arXiv preprint arXiv:1607.08805*, 2016.
- R. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 630–631. Society for Industrial and Applied Mathematics, 2005.
- N. Korula and M. Pál. Algorithms for secretary problems on graphs and hypergraphs. In *International Colloquium on Automata, Languages, and Programming*, pages 508–520. Springer, 2009.
- A. Krause and D. Golovin. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems*, 3(19):8, 2012.
- O. Lachish. $O(\log \log \text{rank})$ competitive ratio for the matroid secretary problem. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 326–335. IEEE, 2014.
- D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov chains and mixing times*. American Mathematical Soc., 2009.
- J. Pearl. Causality: models, reasoning and inference. *Econometric Theory*, 19:675–685, 2003.

A Deferred Proofs

Lemma A.1. $P_{\text{sp}}(n)$ is monotonic non-increasing.

Proof. Let $q := N_{\text{sp}}(n)$. For any n , we have, from the optimality of q :

$$P_{\text{sp}}(n, q) \geq P_{\text{sp}}(n, q-1). \quad (17)$$

From the definition of P_{sp} , we have

$$P_{\text{sp}}(n, q-1) = \frac{q-2}{n} \sum_{i=q-1}^n \frac{1}{i-1} = \frac{q-2}{q-1} P_{\text{sp}}(n, q) + \frac{q-2}{n} \frac{1}{q-2} = (1 - \frac{1}{q-1}) P_{\text{sp}}(n, q) + \frac{1}{n}.$$

Therefore

$$0 \leq P_{\text{sp}}(n, q) - P_{\text{sp}}(n, q-1) = \frac{1}{q-1} P_{\text{sp}}(n, q) - \frac{1}{n}, \quad (18)$$

where the first inequality follows from Eq. (17).

Now, from the definition of $P_{\text{sp}}(n, q)$ we have

$$P_{\text{sp}}(n-1, q) = \frac{q-1}{n-1} \sum_{i=q}^{n-1} \frac{1}{i-1} = \frac{n}{n-1} P_{\text{sp}}(n, q) - \frac{q-1}{n} \frac{1}{n-1} = (1 + \frac{1}{n-1}) P_{\text{sp}}(n, q) - \frac{q-1}{n} \frac{1}{n-1}.$$

Therefore

$$P_{\text{sp}}(n-1, q) - P_{\text{sp}}(n, q) = \frac{1}{n-1} P_{\text{sp}}(n, q) - \frac{q-1}{n(n-1)} = \frac{q-1}{n-1} (\frac{1}{q-1} P_{\text{sp}}(n, q) - \frac{1}{n}) \geq 0,$$

where the last inequality follows from Eq. (18). It follows that $P_{\text{sp}}(n-1, q) \geq P_{\text{sp}}(n, q)$. Since $q := \arg\max_t P_{\text{sp}}(n, t)$, this implies that $P_{\text{sp}}(n-1) \equiv \max_t P_{\text{sp}}(n-1, t) \geq \max_t P_{\text{sp}}(n, t) \equiv P_{\text{sp}}(n)$. Therefore $P_{\text{sp}}(n)$ is monotonic non-increasing. \square